T02T90 06262860

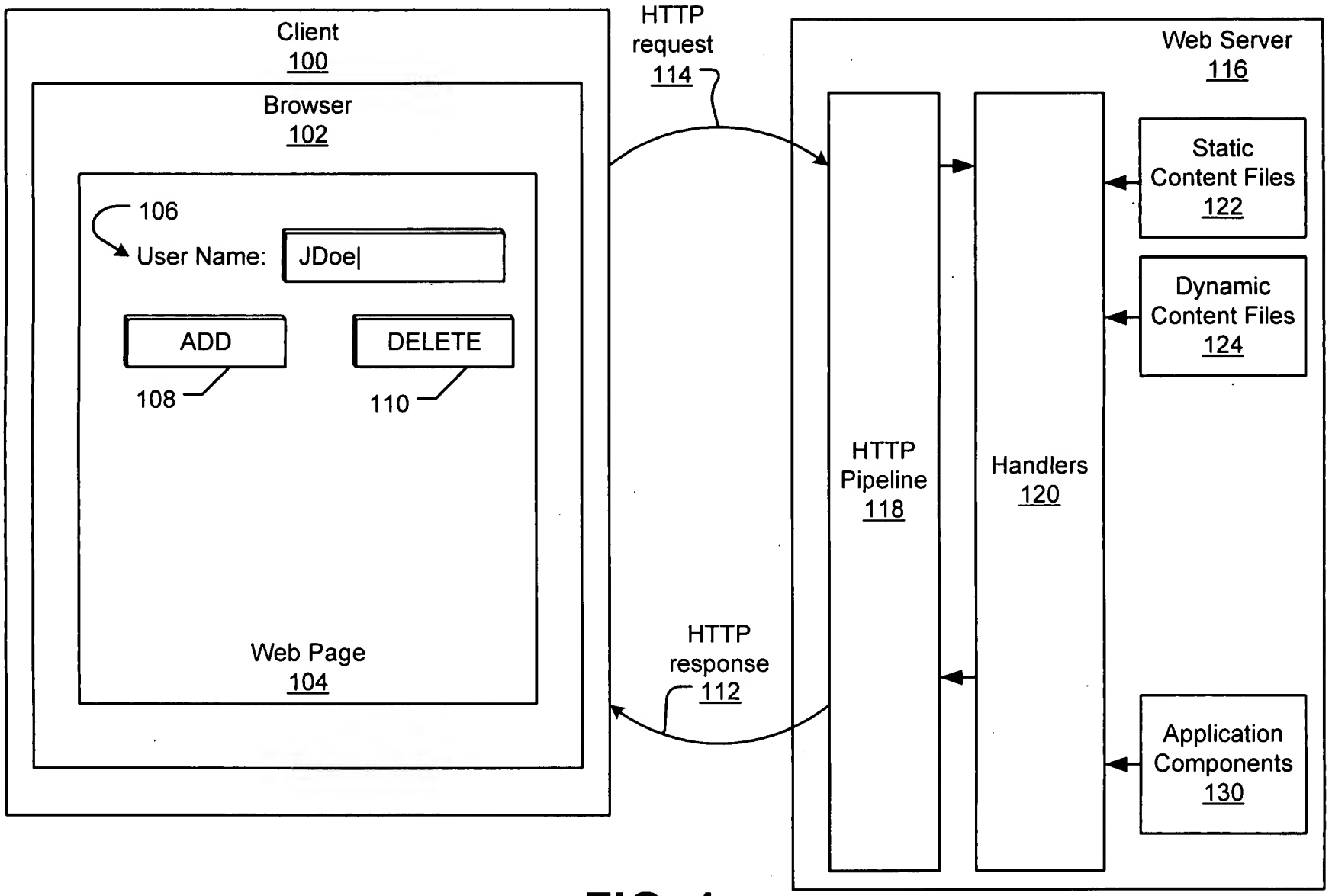


FIG. 1

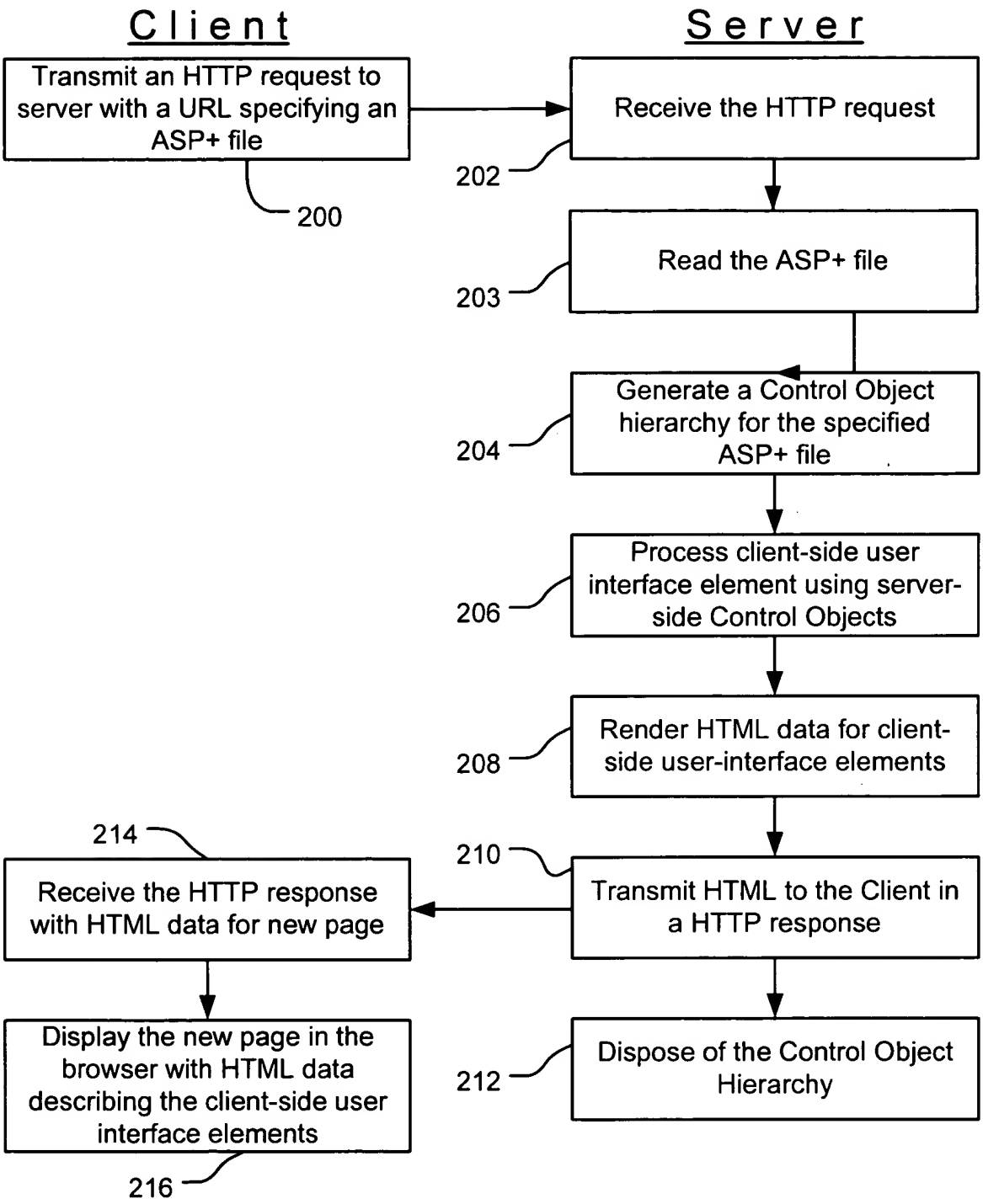

Client
Server
Transmit an HTTP request to server with a URL specifying an ASP+ file
200
Receive the HTTP request
202
Read the ASP+ file
203
Generate a Control Object hierarchy for the specified ASP+ file
204
Process client-side user interface element using server-side Control Objects
206
Render HTML data for client-side user-interface elements
208
Transmit HTML to the Client in a HTTP response
210
Dispose of the Control Object Hierarchy
212
Receive the HTTP response with HTML data for new page
214
Display the new page in the browser with HTML data describing the client-side user interface elements
216

FIG. 2

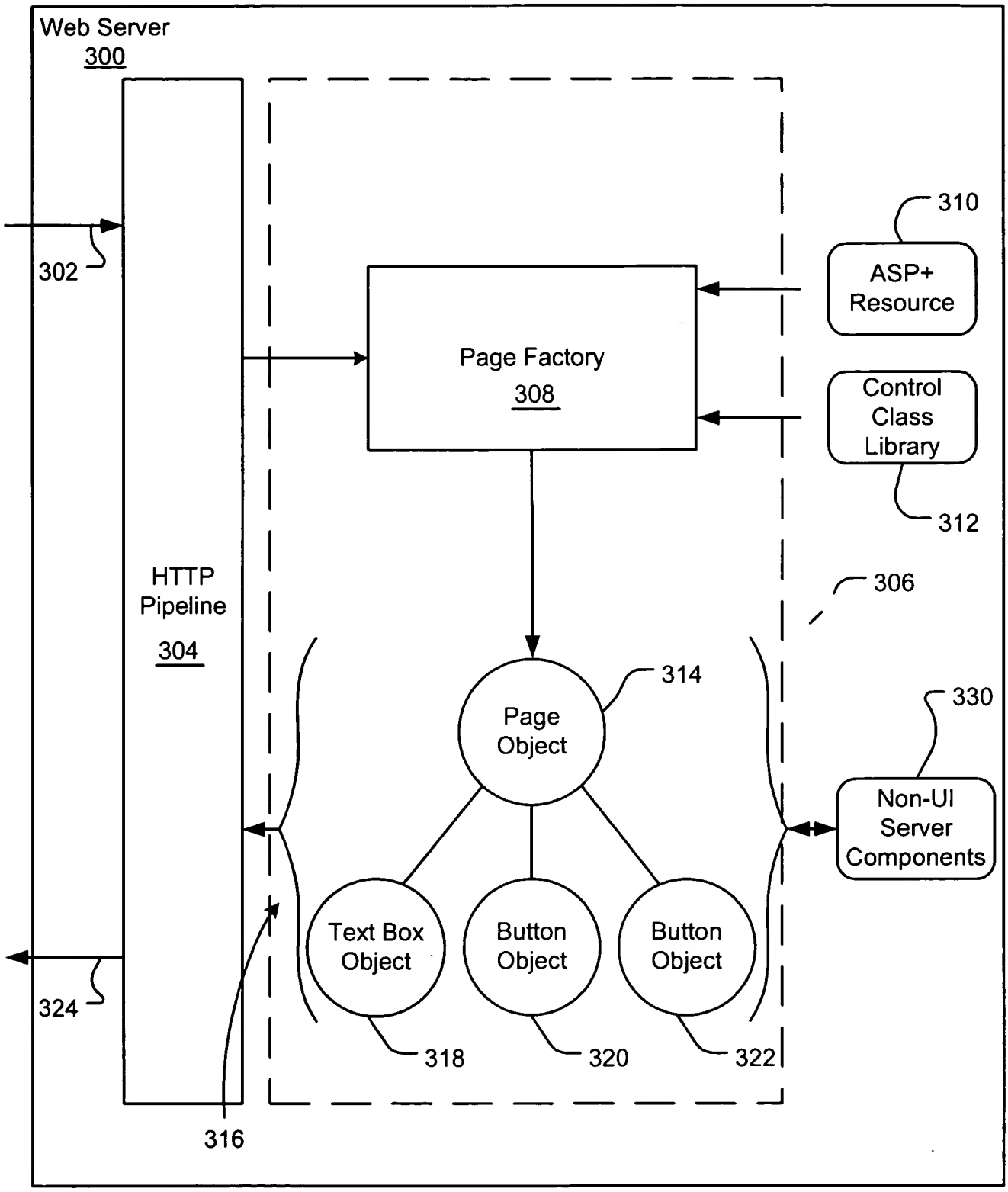

Web Server
300
302
HTTP
Pipeline
304
Page Factory
308
310
ASP+
Resource
Control
Class
Library
312
306
Page
Object
314
330
Non-UI
Server
Components
Text Box
Object
Button
Object
Button
Object
318
320
322
324
316

FIG. 3

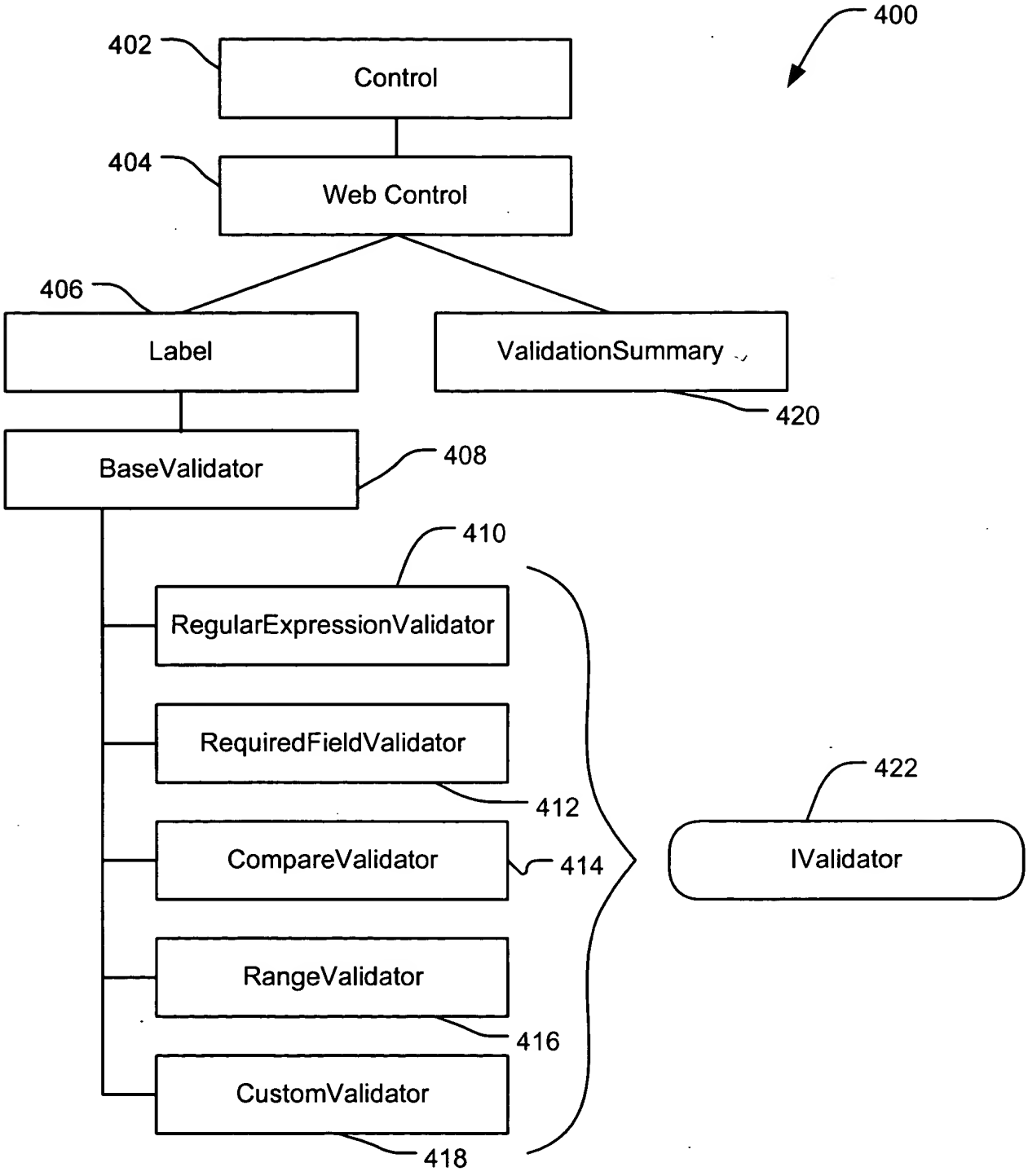

400
402
Control
404
Web Control
406
Label
ValidationSummary
420
BaseValidator
408
410
RegularExpressionValidator
RequiredFieldValidator
412
CompareValidator
414
RangeValidator
416
CustomValidator
418
422
IValidator

FIG. 4

500

```
(1)   namespace System.Web.UI
(2)   {
(3)         public interface IValidator
(4)         {
(5)               // Properties
(6)               public Boolean    IsValid          {get;set;}
(7)               public String     ErrorMessage     {get;set;}
(8)
(9)               // Methods
(10)              public void       Validate();
(11)        }
(12)  }
```

FIG. 5

600

```
(1)   namespace System.Web.UI
(2)   {
(3)         [AttributeUsage(AttributeTargets.Class)]
(4)         public class ValidationProperty : Attribute
(5)         {
(6)               // Properties
(7)               public String          ValidationProperty   {get;}
(8)
(9)               // Constructor
(10)              public          ValidationProperty(String propertyName);
(11)        }
(12)  }
```

FIG. 6

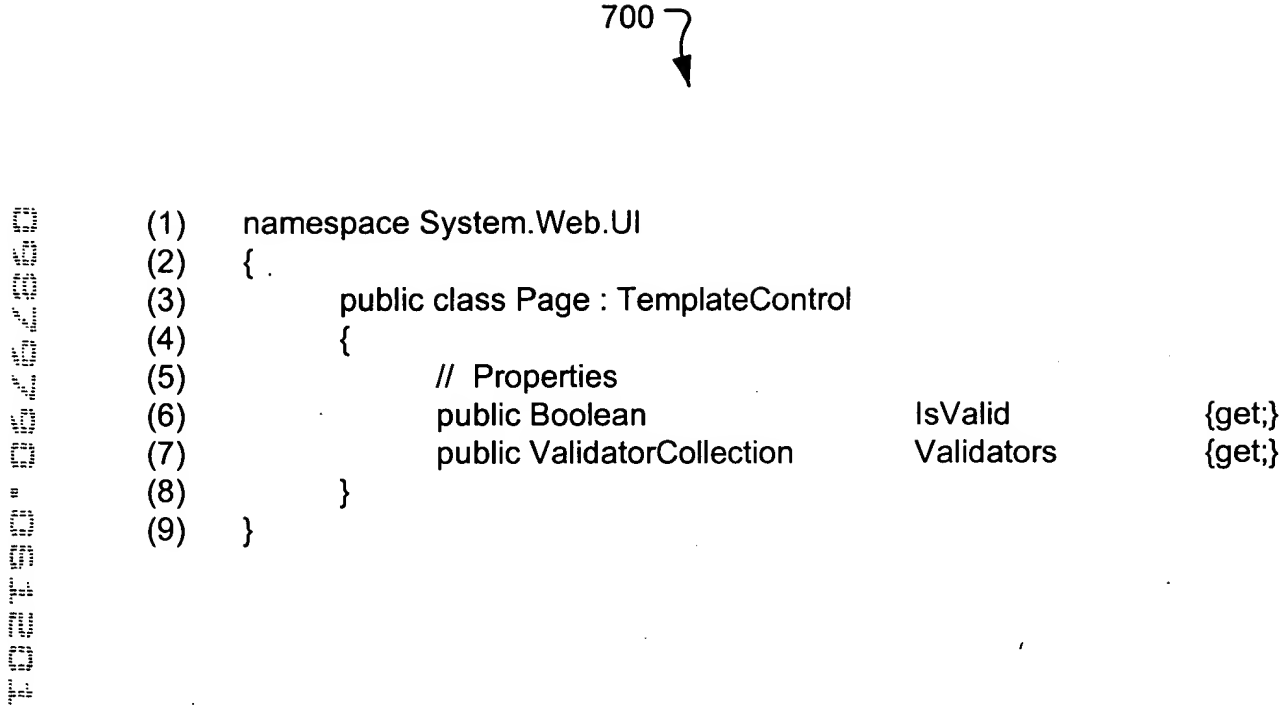

700
(1) namespace System.Web.UI
(2) {
(3) public class Page : TemplateControl
(4) {
(5) // Properties
(6) public Boolean IsValid {get;}
(7) public ValidatorCollection Validators {get;}
(8) }
(9) }

FIG. 7

```
(1)   namespace System.Web.UI
(2)   {
(3)         public class BaseValidator : Label, IValidator
(4)         {
(5)               // Properties
(6)               public String              ControlToValidate   {get; set;}
(7)               public Boolean             IsValid             {get; set;}
(8)               public String              ErrorMessage        {get; set;}
(9)               public ValidatorDisplay    Display             {get; set;}
(10)              public String              Text                {get; set;}
(11)              Public Boolean             EnableClientScript  {get; set;}
(12)
(13)              // Methods
(14)              public void                Validate();
(15)        }
(16)
(17)        public enum ValidatorDisplay
(18)        {
(19)              None = 0;     // Validator is always invisible; used when the error
(20)                            // message is only shown in a ValidationSummary
(21)              Static = 1;'  // Validator takes up space even when hidden
(22)              Dynamic = 2;// Validator takes up space only when visible
(23)        }
(24)  }
```

FIG. 8

800

```
(1)   namespace System.Web.UI
(2)   {
(3)         public class RegularExpressionValidator : BaseValidator
(4)         {
(5)               // Properties
(6)               public String        ValidationExpression       {get;set;}
(7)         }
(9)   }
```

FIG. 9

1000

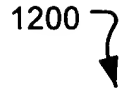
```
(1)   namespace System.Web.UI
(2)   {
(3)         public class RequiredFieldValidator : BaseValidator
(4)         {
(5)               // Properties
(6)               public String        InitialValue   {get;set;}
(7)         }
(9)   }
```

FIG. 10

1100

```
(1)   namespace System.Web.UI
(2)   {
(3)         public class CompareValidator : BaseValidator
(4)         {
(5)               // Properties
(6)               public ValidationDataType          Type           {get; set;}
(7)               public ValidationCompareOperator   Operator       {get; set;}
(8)               public String                      ControlToCompare {get; set;}
(9)               public String                      ValueToCompare   {get; set;]
(10)
(11)        }
(12)        public enum ValidationComparisonOperator
(13)        {
(14)              Equal = 0,
(15)              NotEqual = 1,
(16)              GreaterThan = 2,
(17)              GreaterThanEqual = 3,
(18)              LessThan = 4,
(19)              LesserThanEqual = 5,
(20)              DataTyhpeCheck = 6,
(21)        }
(22)        public enum ValidationDataType
(23)        {
(24)              String = 0,
(25)              Integer = 1,
(26)              Double = 2,
(27)              Date = 3,
(28)              Currency = 4,
(29)        }
(30)  }
```

FIG. 11



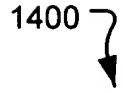

```
(1)   namespace System.Web.UI
(2)   {
(3)         public class RangeValidator : BaseValidator
(4)         {
(5)               // Properties
(6)               public ValidationDataType      Type  {get;set;}
(7)               public String                  MinimumControl   {get;set;}
(8)               public String                  MinimumValue     {get;set;}
(9)               public String                  MaximumControl   {get;set;}
(10)              public String                  MaximumValue     {get;set;}
(11)        }
(12)  }
```

FIG. 12

1300

```
(1)   namespace System.Web.UI
(2)   {
(3)         public class CustomValidator : BaseValidator
(4)         {
(5)               // Properties
(6)               public String          ClientValidationFunction    {get;set;}
(7)               public Event void      OnServerValidationFunction(Object Source,
                        ServerValidateEventArgs Value)          {get; set;}
(8)         }
(9)
(10)        public class ServerValidateEventArgs: EventArgs
(11)        {
(12)              public ServerValidateEventArgs(string value, Boolean isValid);
(13)              public String  Value {get;}
(14)              public Boolean IsValid {get;set;}
(15)        }
(16)  }
```

FIG. 13




```
(1)   namespace System.Web.UI
(2)   {
(3)         public class ValidationSummary : WebControl
(4)         {
(5)               // Properties
(6)               public String                          HeaderText      {get; set;}
(7)               public ValidationSummaryDisplayMode    DisplayMode     {get; set;}
(8)               public Boolean                         ShowSummary     {get; set;}
(9)               public Boolean                         ShowMessageBox  {get; set;}
(10)        }
(11)        public enum  ValidationSummaryDisplayMode
(12)        {
(13)              List = 0;
(14)              BulletList = 1;
(15)              SingleParagraph = 2;
(16)        }
(17)  }
```

FIG. 14

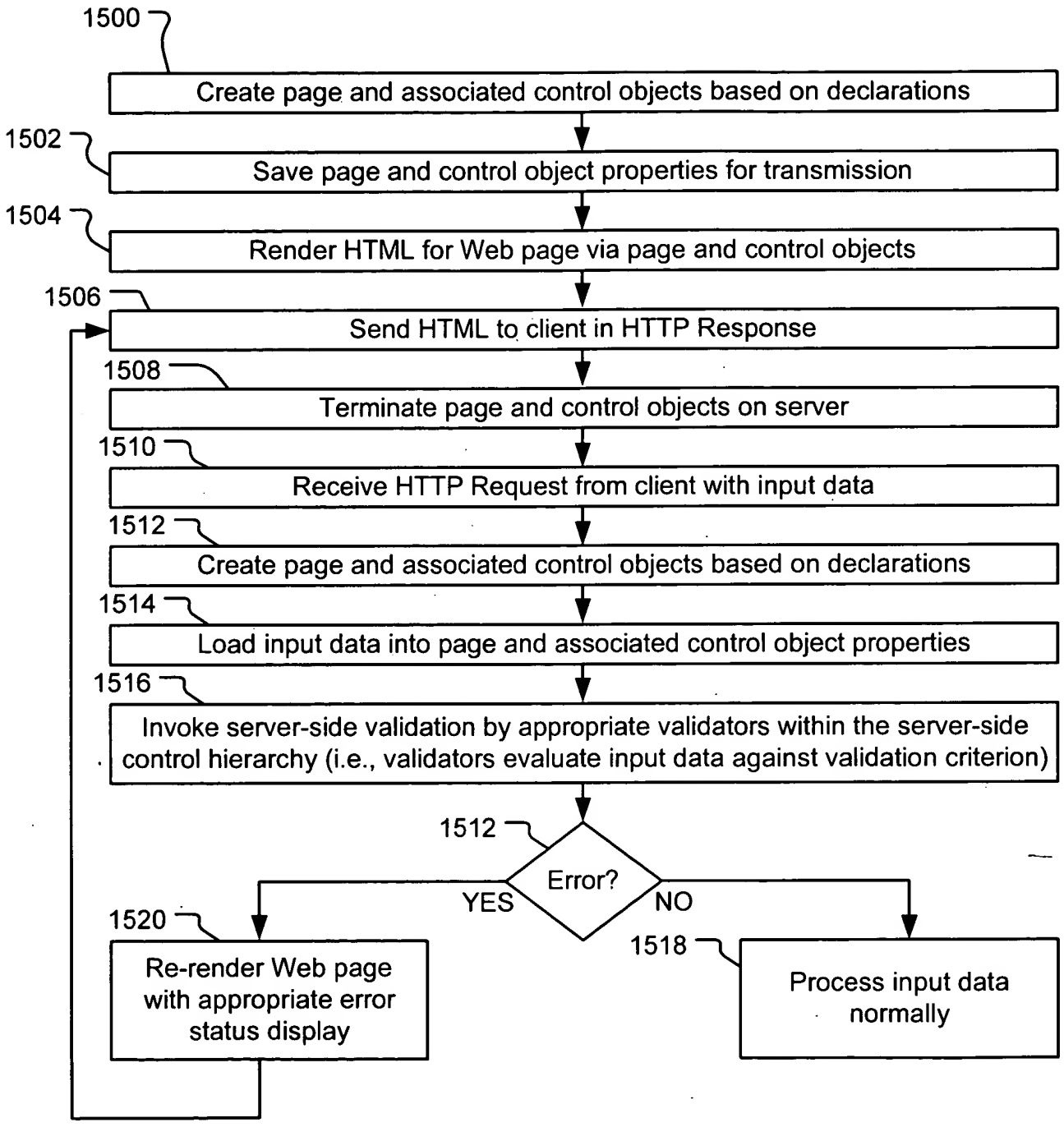

1500
Create page and associated control objects based on declarations
1502
Save page and control object properties for transmission
1504
Render HTML for Web page via page and control objects
1506
Send HTML to client in HTTP Response
1508
Terminate page and control objects on server
1510
Receive HTTP Request from client with input data
1512
Create page and associated control objects based on declarations
1514
Load input data into page and associated control object properties
1516
Invoke server-side validation by appropriate validators within the server-side control hierarchy (i.e., validators evaluate input data against validation criterion)
1512
Error?
YES
NO
1520
Re-render Web page with appropriate error status display
1518
Process input data normally

FIG. 15

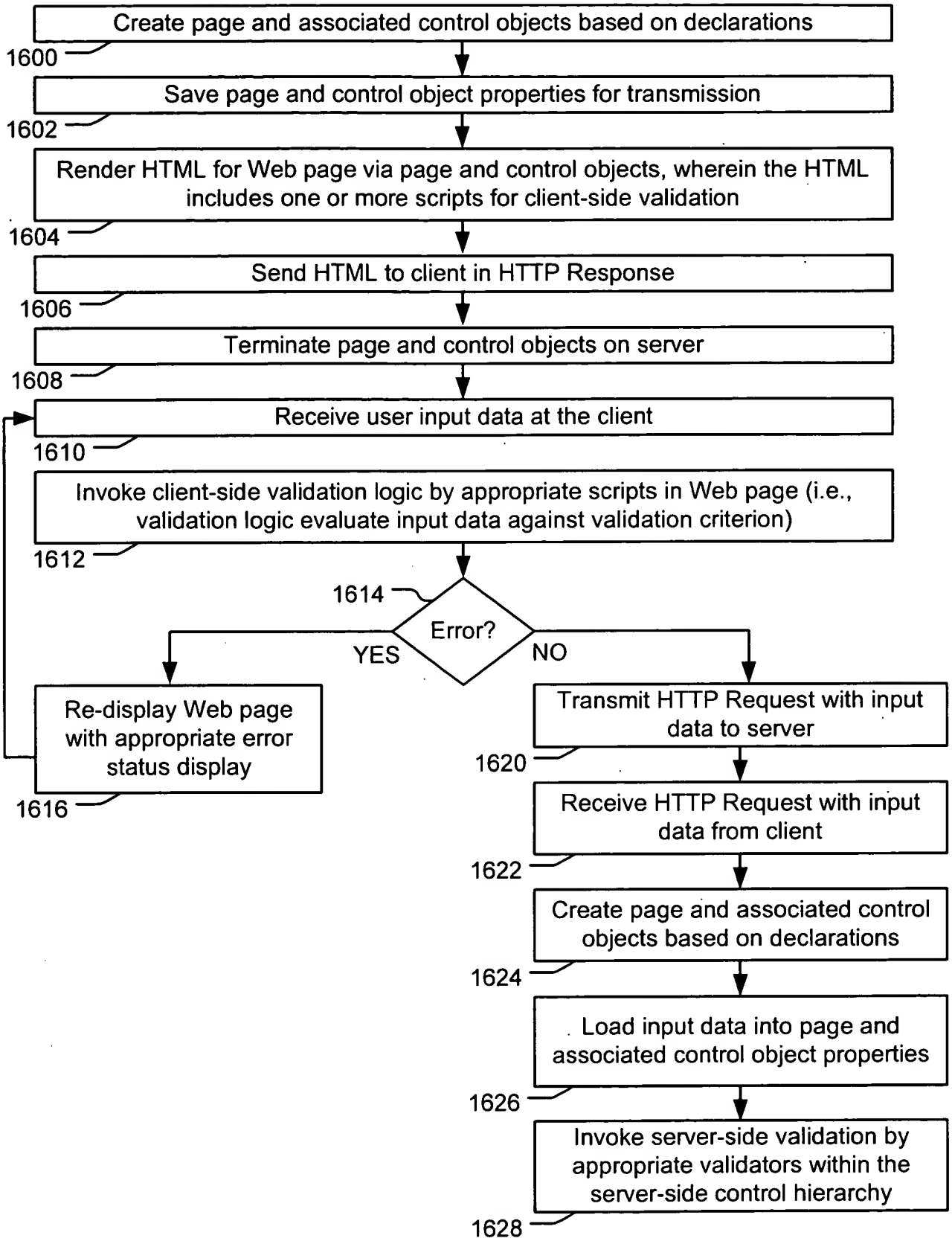

Create page and associated control objects based on declarations
1600
Save page and control object properties for transmission
1602
Render HTML for Web page via page and control objects, wherein the HTML includes one or more scripts for client-side validation
1604
Send HTML to client in HTTP Response
1606
Terminate page and control objects on server
1608
Receive user input data at the client
1610
Invoke client-side validation logic by appropriate scripts in Web page (i.e., validation logic evaluate input data against validation criterion)
1612
1614
Error?
YES
NO
Re-display Web page with appropriate error status display
1616
Transmit HTTP Request with input data to server
1620
Receive HTTP Request with input data from client
1622
Create page and associated control objects based on declarations
1624
Load input data into page and associated control object properties
1626
Invoke server-side validation by appropriate validators within the server-side control hierarchy
1628

FIG. 16

T02T90" 06262860

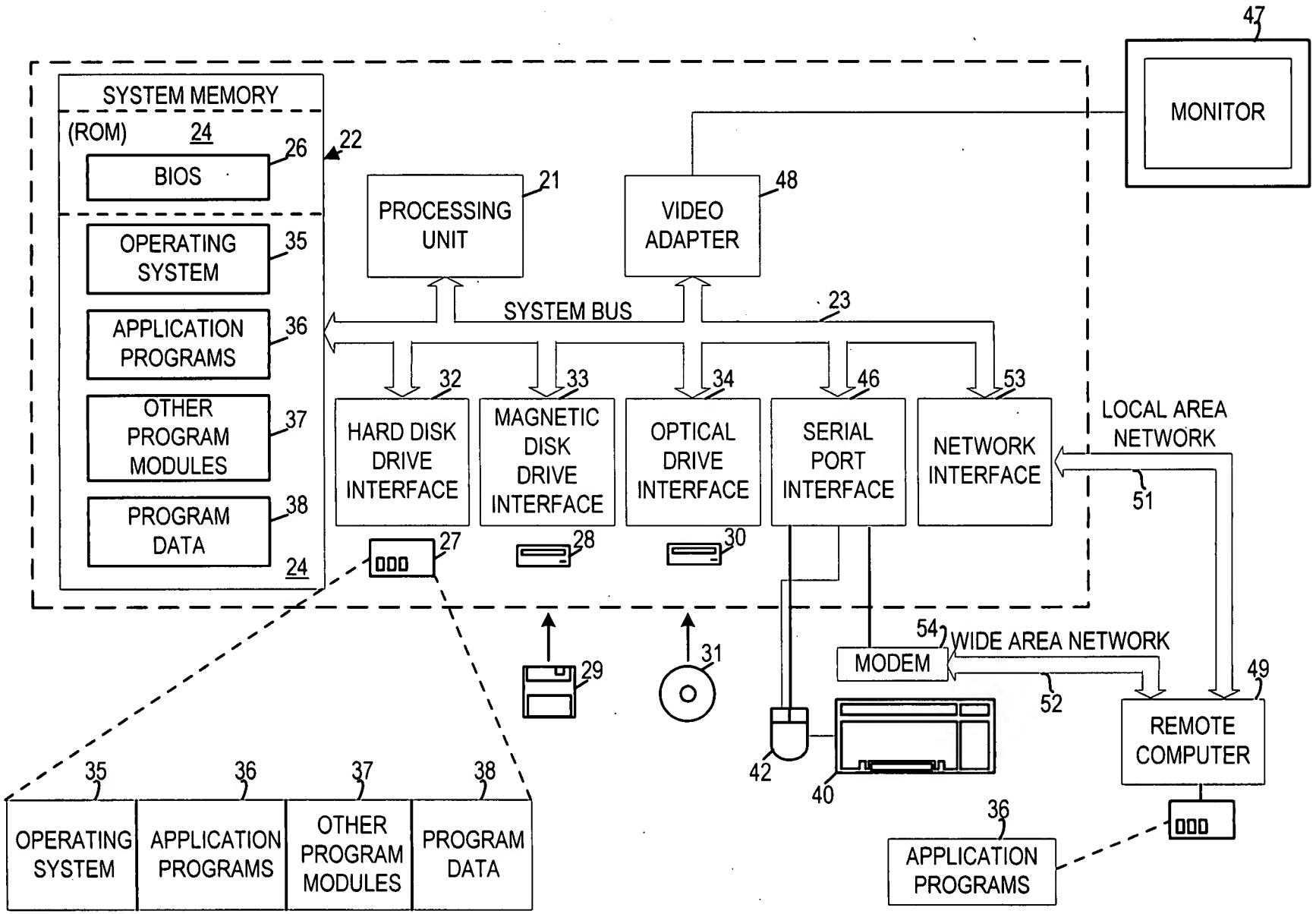



FIG. 17